k-Anonymity using Two Level Clustering

Manish Verma
211CS2064

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela – 769 008, India

k-Anonymity using Two Level Clustering

in partial fulfillment of the requirements
for the degree of

Master of Technology
in
Computer Science & Engineering
by

Manish Verma
(Roll 211cs2064)
under the supervision of
Prof. Korra Sathya Babu

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela – 769 008, India

Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, India. www.nitrkl.ac.in

Mr. Korra Sathya Babu
Assistant Professor

June 1, 2013

Certificate

This is to certify that the work in the thesis entitled *Privacy Preserving Data Publishing* by *Manish Verma*, bearing roll number 211CS2064, is a record of an original research work carried out by him under my supervision and guidance in partial fulfillment of the requirements for the award of the degree of *Master of Technology* in *Computer Science and Engineering*. Neither this thesis nor any part of it has been submitted for any degree or academic award elsewhere.

Prof. Korra Sathya Babu

Acknowledgment

First of all, I would like to express my deep sense of respect and gratitude towards my supervisor **Prof Korra Sathya Babu** who has been the guiding force behind this work. I want to thank him for introducing me to the field of Privacy Preserving Data Publishing and giving me the opportunity to work under him. His undivided faith in this topic and ability to bring out the best of analytical and practical skills in people has been invaluable in tough periods. Without his invaluable advice and assistance it would not have been possible for me to complete this thesis. I am greatly indebted to her for his constant encouragement and invaluable advice in every aspect of my academic life. I consider it my good fortune to have got an opportunity to work with such a wonderful person.

I wish to thank all faculty members and secretarial staff of the CSE Department for their sympathetic cooperation.

Manish Verma

Abstract

Data publising is becoming popular because of its usage and application in many fields. But original data have some sensitive information of individual whose personal privacy can be violated if original data is published . There are some agreements and policies which have to be fulfilled before publishing data . The techniques or protcols which preserve the privacy and retain useful information to apply data mining is nown as it is known as privacy preserving data publishing.

k-anonymity is a technique to preserve privacy of individual while publishing data which still have useful information to apply data mining. To achieve k-anonymity local recoding algorithms gives less information loss but their execution time is more compared to global recoding algorithms. Their execution time mostly depends on for each cluster how they find the most suitable cluster to merge it,its linear search takes unnecessary time which can be reduce by find some most suitable cluster without linear search which we applied in our purposed algorithm. In our work, we used clustering at two levels , cluster at outer level contains inner clusters which are most likely to be merged. so to satisfy k value ,inner clusters merge within same outer cluster if still it do not satisfy k-anonymity then they merge with inner clusters of some other outer cluster, which other outer cluster is most suitable can be find without linear search and most of its inner cluster which still unsatisfied k-anonymity can be find without linear search. It this way we have reduced the execution time of our algorithm which it lesser than other efficient local recoding algorithm KACA and TopDown -KACA and other metrics such as distortion and discernibility gives similar resulted value as other local recoding algorithms.

Keywords: k-Anonymity,Data Privacy, Local Recoding Algorithm, Data Anonymity,

Contents

List of Figures

List of Tables

Chapter 1

Introduction

From the last two decade, the demand of data collection by the individual , government, corporate has been increasing continuously. Because of mutual benefits, data needs to be published .But original data have some sensitive information of individual whose personal privacy can be violated if it is original data is published. There are some agreements and policies how data should be published , it is known as privacy preserving data publishing.

There are many advantage of data publishing as it can be used to understand business trends or patterns to take critical decision .For example to improve the accuracy of recommendations of movie, Netflix a movie rental service has published 500,000 subscribes of movie rationg. It can be used in research field or in medical-record of patients also by applying the techniques of data mining. For example, in California the licensed hospitals submits the demographic data record of their discharged patients . In original data, it contains some sensitive information of a person while publishing it in original form it leads to violation of privacy of that individual .So some policies and agreement have to be followed before publishing the data of individuals. The disadavantage of this approach is either there will be some data loss or a highly trust is required which is impossible in most of the data publishing scenario.

So the challenge task is to generate techniques and tools which are reliable for

data publishing even in hostile conditions also. So while publishing data , the privacy is also preserved is known as Privacy Preserving Data Publishing.

1.1 Privacy-Preserving Data Publishing

For publishing the data a scenario is explained in Figure .For Data collection phase , From record owners , data is collected by data publisher and In data publish phase ,collected data is released by data publisher for data analysis or publicly release it. The person who will apply data mining techniques to gain some knowledge on released data is data recipient .Here data publisher is hospital ,who collects raw or original data from its patient and release for medical center for research purpose.so here medical centre is data recipient.On this patient records any data mining technique can be applied .

For publishing data there are two models

1. UnTrusted Model : In this model , data publisher cannot be trusted so original sensitive information cannot be given to it. Some cryptographic techniques [2] and anonymous methods [1] are proposed sothat anonymously records or information can be collected by the record owner.

2. Trusted Model : in this model , data publisher can be trustful so record owner can directly give their personal details to it. Though there will be issues of privacy while publishing data in data publishing phase.

The non-expert data publisher: In this ,while publishing data, it there is no need of knowledge of data mining for data publisher . Only data recipient will do all data mining operations. As we have explained the example of hospitals in California In that case ,it give data publisher here it is hospital just anonymized the data and give it to medical research center for apply data mining.In this case to get better result by data mining on data, data publisher publish data with preserving some

specific pattern so that it will be helpful for data recipient to apply operation of data mining.

It is very risky to trust the data recipient as we have seen in that case the data recipient is a medical research Centre so it could be risky to trust all the employee of it so the major challenge here is to preserve the privacy while publishing the data .Data publishing deals with publishing of data only it does not association rule mining. Data must be truthful at record level. There are some cases it is important that for each record for the published data there must be some real exist of that entity or person. As we have discussed in case medical research center , if record published does not have real existence, if researcher data recipient, here it is pharmaceutical wants to refer the previous medical condition of patient, in that case the result of data mining would not be meaningful or inappropriate.

1.2 Anonymization Approach

In the most basic PPDP approach, the data publisher has a table of the form D(Explicit Identifier, Quasi-Identifier, SensitiveAttributes, Non-SensitiveAttributes), where Explicit Identifier is attributes that explicitly points to the individual for its record in the table eg name, voter id .

1.2.1 Quasi-identifers

A set of attributes from a table whose combination can be used to identify some other record from dataset. Quasi-identifers may be used to identify any individual record from the table. For example combination of (Job ,Postcode,,BirthDate) combination of all these attribute may used to identify any individual record from the table, to his/her medical problem. Equivalence Class: From table 1.2, with respect to all quasi-identifier whose same values for all the records in the table ,is known as a Equivalence Class. eg.(cat1,*,4350), (cat1,1955,5432), (cat2,1975,4350) are the three equivalence from the generalized table.

1.2.2 *k*-Anonymity

From a generalized table,Number of records present in every equivalence class gives the size of equivalence class, with respect to all attribute set Q, .For all equivalence class wrt Q in a table must have size atleast *k*. Eg. As shown in table 1.2, For 2-anonymization, all three equivalence class have size 2. As the size of k increase better will be privacy level.

1.2.3 Senstive Attribute

Sensitive Attributes contain the sensitive person-specific information with information do not want to tell to others, such as disease, salary and disability status . Non-Sensitive Attributes contains all attributes that do not fall into the previous three categories.

Table 1.1: Original Table

Job	**Birth**	**Postcode**	**illness**
Cat1	1975	4350	HIV
Cat1	1955	4350	HIV
Cat1	1955	5432	flu
Cat1	1955	5433	flu
Cat2	1975	4350	flu
Cat2	1975	4350	fever

[4]

1.3 Anonymization

Releasing data Publicly of any individual, it might be risk to their privacy, a recent survey done by L seweney[1] explained 87 percentage population of USA can be individual identified by taking the three attribute data Age , Date of Birth , Zipcode.

Table 1.2: 2- Anonymized Table [4]

Job	**Birth**	**Postcode**	**illness**
Cat1	*	4350	HIV
Cat1	*	4350	HIV
Cat1	1955	543*	flu
Cat1	1955	543*	flu
Cat2	1975	4350	flu
Cat2	1975	4350	fever

He showed by taking these three attributes how Willam Weld, the governor of USA can be identified. It is very easy to collect these three attributes of any person.

In this example, by linking the quasi-identifier of record owner his identity can be re-identified. For this attack only two prior knowledge must be known first is victim s record must be present in the published table and his original values for its quasi-identifier.

This attack can be prevented when data publisher publish an anonymized table [11, 17] T(QID' , Sensitive Attributes, Non-SensitiveAttributes),

QID' is ananonymized form of the original QID generated by applying anonymization functions to the attributes in QID in the original data table D. Anonymization technique hides the information of few quasi-identifier that some other records also become similar to that record in that table .If a person if identified by his record for that record there must be some other person whose records are similar to this record entry. In anonymization technique some noise is added to original data so that it can fulfill the all the conditions which are necessary for that privacy model. There are some metric which can be used to measure the quality of anonymized data. In this model, non a sensitive attribute published they can used for data mining.

1.4 Attack Models and Privacy Models

A specific definition of privacy preservation is that from the published data , there must not be any extra information gained by attacker by apply the data-mining on published data. But in reality it is shown by Dwork [12, 15, 16] that it can not be completely achieved because there attacker also knew some background knowledge of target victim. The attack principles classifies the privacy model in mainly two categories. In the first category, if attacker can map a person record to the record which is present in published data, and its corresponding sensitive attribute , it is known as linkage attribute, in this quasi identifier if victim is known . In second category, [13,18]attacker gains the more information about the victim by using the background knowledge prior known to him .If there will be huge difference between prior and posterior beliefs of attacker, it is known as probabilistic attack.

1.4.1 Record Linkage

In this attack, a few number of records maps in the released table based on the quasi-identifier matched to the quasi-identifier of the target victim .Based on the background knowledge about victim it may be uniquely identified in this case.

In table (a) to medical center .By referring to records from table 1.3a , The research center maps the records based on same quasi-identifiers present in both table it gain sensitive information , here by joining these two tables 1.3a and 1.3b for quasi -identifier job, sex and age it can found that male whose age is 38 and profession is lawyer suffers from HIV is mapped to Doug.

To avoid such type of attack by record linkage , a new technique is proposed by Sweeney ,Samrati [14, 19] in this model for each set of all quasi-identifiers having same value in table must have atleast k number of records .The benefit of this model is that that there other k-1 records with maps to same quasi-identifier set of the probability of attack becomes 1/k. As it shown in table 1 for quasi-identifier (job,birth,postcode).

Subset Property of K anonymity
If a table is k anonymous with a set of quasi-identifiers Q , then the must satisfy k anonymity with respect to all subset Q [20, 21, 25].

(X,Y)-Anonymity The assumption of k anonymity is that each records present in anonymized table is unique existence in real life which may not be true for example let a patient may have more than one disease at a time so it might be possible it its quasi-identifier present in original table may satisfy k but in reality their records links to single identity.To avoid this problem [28] proposed (X,Y)-anonymity, where X and Yare disjoint sets of attributes. A_Y (X) is the anonymity for set of quasi-identifiers X .it is the total number of unique Y values with respect to same X. So the table satisfy (X,Y) anonymity if A_Y (X) $\geq$ K.

It states that for set of attribute size(quai-identifier) X must be mapped to at least Y unique values. Eg. as in previous case ,X is set of {Job,Sex,Age} and Y is the sensitive attribute so for each same set of X there must be at least Y different values.

1.4.2 Attribute Linkage

In this attack , attacker gain some information about his sensitive attribute from the released table , even though attacker is not able to link the victim with any individual published record .From the table 1.3d, attacker can find that all the female having age 30 whose profession is dance suffer from HIV.so {Dance,Female ,30} is confidence 100 percent HIV by this information it found that Emily suffers from HIV. L -Diversity. To prevent from attribute linkage attack it is purposed by Machanavjjhala [13] .Its necessary conditions is every equivalence of released table must have at least l different values.The fundamental concept is to avoid attribute linkage as we seen from the last example if there will be different unique sensitive values it prevents attribute linkage. But probabilistic attacks can not be avoided by this because flu is very common disease compared to HIV.The released table is l-diverse if for all qid group.

$$-\sum P(qid, s)log(P(qid, s)) \geq log(l) \tag{1.1}$$

Here S is sensitive attribute, P(qid,s) is fraction of records whose sensitive value is s for the total records whose equivalence class is group denoted by qid . The more uniformly distributed sensitive values in each equivalence class group qid higher will be the entropy of sensitive attribute. So higher value of entropy in the released table , lesser is the chances probabilistic attack, higher value of threshold l increases its privacy and lesser is the information gain by attacker from released table.

Limitations The major limitation of entropy l -diversity is it can not the measure of probabilistic attack for eg as it is calculated entropy is 1.8 but in second equivalence group out of 4 records 3 suffers from HIV from table 1.3d, which is easy for probabilistic attack.

1.5 Anonymization Operations

The table which contains the original records values of each individual person do not provide any privacy. To publish it and to preserve the privacy of each individual person, some operations have to be performed . Anonymization is a technique to solve the problem of data publishing, it while keep the sensitive information of record owner which is to be used for data analysis it hides the explicit identity of that record owner from the table which is going to be published.

Anonymization can be done by using following operations

1. Generalization
2. Suppresion

1.5.1 Generalization

Generalization modifies the quasi-identifier original most specific value to the some generalized values of specific description, eg specific form date of birth to generalized

Job	Sex	Age	Diease
Engineer	Male	35	Hepatitis
Engineer	Male	38	Hepatitis
Lawyer	Male	38	HIV
Writer	Female	30	Flu
Writer	Female	30	HIV
Dancer	Female	30	HIV
Dancer	Female	30	HIV

(a) Patient table

Job	Sex	Age	Desiese
Professional	Male	35-40	Hepatitis
Professional	Male	35-40	Hepatitis
Professional	Male	35-40	HIV
Artist	Female	30-35	Flu
Artist	Female	30-35	HIV
Artist	Female	30-35	HIV
Artist	Female	30-35	HIV

(b) 3-Anonymous Table

Name	Job	Sex	Age
Alice	Writer	Female	30
Bob	Engineer	Male	35
Cathy	Writer	Female	30
Doug	Lawyer	Male	38
Emily	Dancer	Female	30
Fred	Engineer	Male	38
Gladys	Dancer	Female	30
Henry	Lawyer	Male	30
Irene	Dancer	Female	32

(c) External Table

Name	Job	Sex	Age
Alice	Artist	Female	[30-35)
Bob	Professional	Male	[35-40)
Cathy	Artist	Female	[30-35)
Doug	Professional	Male	[35-40)
Emily	Artist	Female	[30-35)
Fred	Professional	Male	[35-40)
Gladys	Artist	Female	[30-35)
Henry	Professional	Male	[30-35)
Irene	Artist	Female	[30-35)

(d) 4 Anonymous External Table

Table 1.3: k- Aonymity Example

[4]

to year only while hiding month and date value. Full-domain generalization scheme [7–9, 22] while generalizing, for all records and for any quasi-identifier values are

generalized to same level of hierarchy tree For eg. If a equivalence class of {writer, dancer } is generalized to Artist then other equivalence of {Engineer ,Lawyer } must be generalized to Professional. Generaized table is consistent and it is used in Global recoding algorithms, but the major drawback of this is data loss is very high. .

1. Subtree Generalization

   In this generalization scheme [9, 20, 21] , At any non-leaf node either all its child values are generalized or none is generalized. For example from figure if all dancer is generalized to artist then writer have to be generalized to artist but doctor and engineer may be generalized can retain its specific value at leaf level.It is used in Global recoding algorithms.

2. Sibling Generalization

   In this generalization scheme [22], it is same as subtree generalization but in this some sibling can remain ungeneralized . For example if Dancer is generalized to artist then writer can remain ungeneralized . It gives the lesser distortion compared to subtree and full domain and used in global recoding algorithms.

3. Cell Generalization

   All the generalization [23] scheme that are discussed earlier are used are called global recoding. They give more distortion in this scheme is a value is generalized in one record then for that specific value must be generalized in all other records also.

   But In cell generalization, it is known as local recoding there is not restriction means if a value is generalized in one record the same value for same attribute in other record may be ungeneralized. For example in a record dancer is generalized to artist dancer in other records may remain ungeneralized. The problem of this flexibility is that data utility is affected by this because while applying data mining technique in this dancer assign to class 1 and assign to class 2 so both are two different classes. While Global recoding generalizing

scheme donot have this data utility problem.

1.5.2 Suppression

Suppression is similar to generalization but in this values of quasi-identifier is completely hidden for eg from sex male female to Any or not released or from specific profession to value is suppressed to not released at all. Different Supression[22, 24] types are defined as

1. Record Level :When the complete entry of a record from the table is eliminated or suppressed.

2. Value Level : When all instance or records of a particular value in the table is suppressed.

3. Cell Level : When some of records for a given value are suppressed in a table.

1.6 Motivation

1. As we have seen local recoding algorithm execution time mostly depends on how a cluster search the most suitable other cluster to satisfy k-value with minimum distortion or any other metric. Execution time can be decreased if complete dataset is partitioned into some bigger clusters, which contains records which are more likely to be merged so search is done within bigger cluster, means if we increase the number of clusters lesser will be execution time.

2. Inside each bigger clusters, instead of searching linearly for every record , if we can use some mathematical pattern or any other relation so that we can merge records inside bigger cluster with less no of time linear search, it will decrease execution time.

1.7 Objective

1. To implement clustering at two level ,find the cluster number at both level for each record sothat assign it to a outer and inner cluster based on its equivalence class.

2. To find without linear search for each outer cluster which inner cluster are more likely to merge and for inner clusters of any outer cluster which inner cluster of other outer cluster are more likely to merge without linear search.

1.8 Thesis Organization

Ch 1 Introduction

In this chapter we have discussed briefly about data publishing and what is privacy preserving,why there is need of privacy preserving techniques whiling publishing data. How anonymization can be used to preserve privacy .To maintain privacy a model K anonymity is explained in it and its basic details and attack on this model.

Ch 2 Related Work

In this chapter we have discussed ,metric that are used to calculate the quality of anonymized data , global and local recoding algorithm. we explained the local recoding algorithm and how metrics are used for better anonymization of data.

Ch 3 Motivation

In this chapter we have discussed why local recoding are important and their issues why they take more time to execute and execution time depends upon which factor and how we can resolve the issue and reduce time complexity. **Ch 4 Purposed Work**

In this chapter we explained that To achieve k anonymity clustering can be done at two level , first cluster is to be searched with in same outer cluster and then search in other outer cluster .Most the execution time depends upon how most suitable cluster is to be searched for every cluster, if we can find it without linear search based upon

some relation between them, it can reduce the execution time while consider other metrics also.

Ch 5. Experiment Results

In the chapter we have plotted the graph ,for different values of k taken executimetime vs quasi-identifer, distortion vs quasi-identifer, Discernibility vs quasi-identifer.we can compare and analysis the results of local recoding algorithms with our purposed algorithm.

Ch 6.Conclusion

In this chapter, we have explained that after comparing the results and analysis we can conclude that our purposed algorithm gives takes less time than other efficient algorithms while other metric also gives better results in maximum cases.

Chapter 2

Literature Review

2.1 Metrics used to Measure the Quality of Generalized Data

Privacy preserving data publishing have two objectives, privacy of individual entity for each record must be preserved and published data must be information which is useful for data mining. So the quality of anonymized data can be measured by data metric which are classified into three categories.

2.1.1 General Purpose Metrics

When data publisher do not know what data recipient want to know or analysis from the published data so data publisher can not focus on any particular data utility .In this case data published is open to all like internet so that data recipient based on their different interest and they do data mining according to their requirement, in this is very obvious that same metric is not good or accurate for different recipients. In this case for better utility of anonymized data ,data publisher choose metric which are more suitable for mostly all data recipients such as ILoss, distortion, discernibility.

1. **ILoss**

   To calculate the data loss while anonymizing the data [27] proposed a data metric known as ILoss.

$$ILoss(V_g) = \frac{|V_g| - 1}{|D_A|} \tag{2.1}$$

   Where $|V_g|$ is total number of children of node .

   $|D_A|$ is the total number of leaf nodes for that attribute having vg as a node.

   If ILoss = 0, means value remains ungeneralized ,same as in original table . It calculates the fraction of leaf nodes that are generalized.

   Example:Let a value is generalized from Lawyer to professional.

   So its $ILoss = \frac{2-1}{4} = 0.25$ After generalization ILoss for any record can calculated as

$$ILoss(r) = \sum (W_i \times ILoss(V_g)) \tag{2.2}$$

   W_i is predefined weight penalty assigned to each quasi-identifer The total for complete generalized table is

$$ILoss(T) = \sum_{r \in T} ILoss(r) \tag{2.3}$$

2. **Discernibility**

   After anonymizing dataset ,each equivalence class has its size that is number of records in it. The size of each equivalence class contributes to the cost anonymization, it can be calculated for complete generalized dataset by this formula, Discernibility Metric [10].

$$DM = |E_i|^2 \tag{2.4}$$

   where E_i is the size of equivalence class .

   minimize Discernability cost leads to less distortion with is desirable requirement for better anonymiztion.

2.1.2 Special Purpose Metrics

If data publisher know for which purpose the published data will be data mined or in which information or pattern data recipient is interested ,so that they can preserve their related information and publish the data according to their requirements .For example if the purpose of data recipient is to model the classification based on a particular attribute in this case generalization must not be done for values whose identification is necessary to assign a class,which is used for their classification .

Classification Metric (CM)

Iyengar[24] purposed a metric to measure the classification error means a record is assigned to a class by assuming that in it a particular class is not majority but in reality that class is not the majority class so, record is assigned to wrong class . There must be some penalty for it or there is a penalty if record is suppressed completely and not assigned to the any class. CM can be calculated by sum of all the penalties of each record, it is normalized by considering total number to records.

$$CM = \frac{\sum_{all\,rows} penalty(row\ r)}{N} \tag{2.5}$$

Arowris penalized if it is suppressed or if its class label class(r) is not the majority class label majority (G)of its group G

$$penalty(row\ r) = \begin{cases} 1 & \text{if } r\, is\, suppressed \\ 1 & \text{if } class(r) \neq majority(G(r)) \\ 0 & else \end{cases} \tag{2.6}$$

Penality can be calculated as if a record is suppressed or it is assigned togroup assume class(r) is major class but actual that class is not the major class.

2.1.3 Trade-off Metrics

Specializing from a general value to a specific value loss some level of privacy but gain some information regarding that attribute which is specialized. Special metric

while anonymizing at final information it may gain sufficient information but might lose so privacy that it is very difficult to do further anonymization. So Trade -off Metrics solve this problem, both information gain and privacy loss are calculated at every iteration of anonymization,so that optimal trade -off can be found for both necessary requirements.

In this trade-off metric [4], for every specialization all records of this group are assigned to its child level group so it gain some information(IG)and as it divides the group size into smaller group there is privacy loss(PL) ,.objective of this metric is to find a specialization whose information gain is maximum for each privacy loss

$$IGPL(s) = \frac{IG(s)}{PL(s) + 1} \tag{2.7}$$

Where $IG(s)$ = Information gain can be decrement of class entropy or decrement of distortion by specialization.

$$PL(s) = avg\{A(QID_j) - A_s(QID_j)\} \tag{2.8}$$

privacylossPL(s) = the average decrease of anonymity over allQIDjthat contain the attribute of s.
$A(QIDj)$ = the anonymity before specializing of attribute j
and As $(QIDj)$ = the anonymity of QID_j after specializing of attribute j.

2.2 Alogrithms for k- Anonymity using Local recoding

2.2.1 Anonymization by Local Recoding in Data with Attribute Hierarchical Taxonomies

After anonymzing the original data set the quality of anonymized data can be calculated by calculating some metrics on anonym zed data eg-Distortion [25], Precision Metric [4], CAVG [10], NCP [5] ,Discerniblity metric [10].

- **Weighted Hierarchical Distance (WHD) [10]**

  Let domain hierarch height is h its domain levels are 1, 2,. . ., h-1, h from the most general to most specific, respectively. Let $w_{j,j-1}$ be a predefined weight between j and j-1 in domain heirarchy. where $2 \leq j \leq h$. From level p to level q, where p¿q, a attribute is generalised the WHD for this generalization is measured as

$$WHD(p,q) = \frac{\sum_{j=q+1}^{p} w_{j,j-1}}{\sum_{j=2}^{h} w_{j,j-1}} \tag{2.9}$$

  here $p > q$, $2 \leq j \leq h$

- **Distortion** Let a tuple $t = \{q_1, q_2, ., q_m\}$ and its generealized tuple $t' = \{q_1{}', q_2{}', q_m{}'\}$,so total number quasi-identifier is m ,In attribute hierarchy ,domain level for qjis denoted as level (q_j).Distortion can be calculated by

$$Distortion(t,t') = \sum_{j=1}^{m} WHD(level(v_j), level(v_j{}')) \tag{2.10}$$

  Let $t_1 \, and \, t_2$ be two tuples .is the closest common generalization for t_1, t_2 is denoted as $t_{1,2}$ for all i.

  Let $t_1 = \{male; young; 4351\}$

  $t_2 = \{female; young; 4352\}$

  So $t_{12} = \{*, young, *\}$

- **Distance between two tuples**

  Let t_1, and t_2 are the two tuples and their closest common generalization is $t_{1,2}$. The distance between the two tuples can be calculated as:

$$Dist(t_1, t_2) = Distortion(t_1, t_{1,2}) + Distortion(t_2, t_{1,2}) \tag{2.11}$$

  Let $t_1 \, and \, t_2$ be two tuples .is the closest common generalization for t_1, t_2 is denoted as $t_{1,2}$ for all i.

  Let $t_1 = \{male; young; 4351\}$

  $t_2 = \{female; young; 4352\}$

So $t_{12} = \{*, young, 435*\}$

$Dist(t_1, t_2) = Distortion(t_1, t_{1,2}) + Distortion(t_2, t_{1,2}) = 1.25 + 1.25 = 2.50$

$Total\ Distortion = (1, 0.0.25) = 1.25$

S+o the distortion of anonymized table can be calculated

$$Distortion(D, D') = \sum_{j=1}^{|D|} Distortion(t, t_i{}') \tag{2.12}$$

Where

$|D|$ is total number of records in table.

t_i is the tuple in original table.

$t_i{}'$ is the tuple in anonymized table.

Explanations Details

KACA (k- anonymity Clustering in Attribute Hierarchy) [10] is the algorithm use local recoding to anonymize the data to achieve k-anonymity. In this algorithm records are assigned to cluster and those cluster whose size is smaller than will have to merge with other clusters to satisfy the k value. cluster (C_1) whose size is less than k find the cluster will find the most suitable cluster to merge is based on the distortion which is already discussed.

So cluster searches other cluster (C_2) whose distortion is minimum to this,Here two case arise.

$$Time\ Complexity = O(n \log n + |E_s| * |E|) \tag{2.13}$$

$|E_s| * |E|$ is takem to merge all equivalence class to satify k anonymity .Its checks all equivalence class to minimize distortion which takes longer time to search the most suitable cluster.

Algorithm 1 Algorithm :k-Anonymisation by Clustering in Attribute Hierarchies (KACA)

1: Generate equivalence classes from the data set
2: **while** there exists an equivalence class of $size < k$ **do**
3: randomly choose an equivalence class C of $size < k$
4: evaluate the pairwise distance between C and all other equivalence classes
5: find the equivalence class C with the smallest distance to C_0
6: generalise the equivalence classes C and C_0
7: **end while**

2.2.2 (α,k)-Anonymity: An Enhanced k-Anonymity Model

In the following k-anonymity example , it is achieve by using local , global , multidimensional recoding algorithms. By analyzing k -anonymized using local recoding of table 1.2 we can find that its first equivalence class both records suffers from HIV. This is the breach for privacy which is risky as sensitive attribute is higjly sensible as in this case .It happened because equivalence class ($cat1, *, 4350$) link to same sensitive attribute.

But the Equivalence class $(*, 1975, 4350)$ is mapped to multiple diseases (i.e.flu and fever). In this algorithm a new term α is introduced to preserve the sensitive relationship . α can be explained as :

After achieving k anonymity, in all equivalence class ,the total number of count of any sensitive attribute divided by total number of records in that equivalence class , so this fractional value must not be more than α.It is fractional limit that equivalence can not exceed to satisfy (α, k) anonymity.

(α, k) anonymity Algorithm Explanation:

Top down algorithm approach is used for this, initially all records are fully generalized. One quasi-identifier is chosen based on following two criteria.

- Quasi-identifer which specialize maximum number of records will be chosen

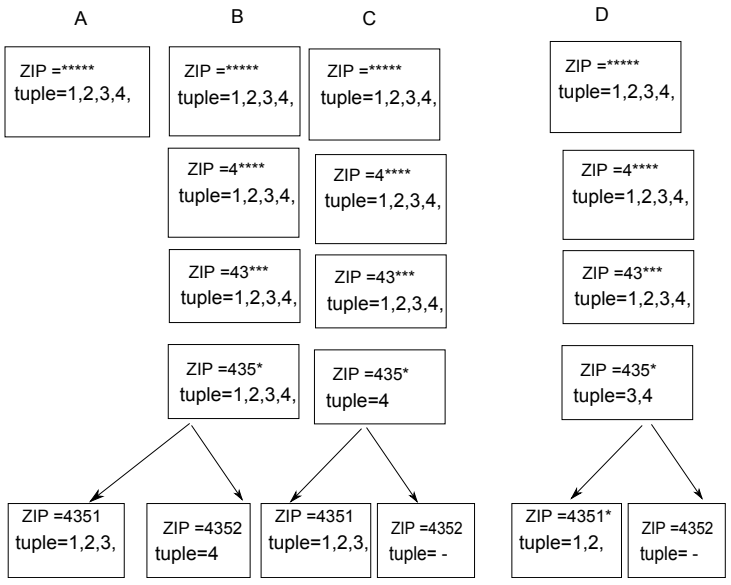

Figure 2.1: Iteration using TopDown Algorithm

for specialization.

- If there will be a tie in first case then quasi-identifier who gives minimum number of branched will be chosen for speacialization.

In this quasi-identifer is chosen for each iteration. When a quasi-identifer is chosen speacialize it in its hierarchy domain until it violates the condition of (α, k) anonymity. The iteration in which it violates this condition, branch that do not fulfill the condition all its records move to its parent branch if that also that fulfill (α, k), records from other branches moved up or again generalize to their parent level so that for all branching (α, k) anonymity condition must be satisfied. It can be seen in figure 2.1 also.

2.2.3 TopDown-KACA: an Efficient Local-Recoding Algorithm for *k*-Anonymity

Topdown Algorithm takes lesser time to execute but their distortion is high while KACA [6]algorithm gives less distortion but it take more time to execute so

TopDown-KACA [3] is the algorithm which uses these two algorithm to reduce execution time ,it partition data into some bigger cluster using TopDown approach and to reduce distortion it use KACA algorithm inside each partition.
So it takes much lesser time than KACA and its distortion is also reduced.
Topdown algorithm [20] and KACA are already explained in this section 2.2.2 and 2.2.1 respectively.

Algorithm 2 Algorithm : TopDown-KACA

1: $D_1, D_2, .., D_m$ =TopDown(D, QI, c)
2: **for** $i \leftarrow 1, m$ **do**
3: E_i = KACA(D_i, QI, k)
4: **end for**
5: if exists an equivalence class E whose size is less than k
6: **for** $i \leftarrow 1, |E|$ **do**
7: insert it into its nearest equivalence class
8: **end for**

Chapter 3

k- Anonymity using Two Level Clustering

3.1 Two level Clustering and their corresponding Equivalence classes

We implemented Clustering at 2 level , Each Outer cluster also contains inner cluster which are more likely to be merge by generalizing one or Quasi-identifier. All inner clusters must have different equivalence class (EQ) ie 0 level EQ class but must have same 1st level Equivalence class for outer cluster.

For every outer cluster there is mathematical relationship between all inner cluster to find which are more suitable to merge for any particular Qi means we can say within each outer cluster which inner cluster are to be merge it can be known without linear search.It will reduce the execute time

At outer level each cluster assign a 1st level Equivalence class and its integer sequence number which is unique for to it , it is used to access this cluster directly instead of linear search.

Similarly , within each outer cluster each inner cluster must have its unique 0 level equivalence class and its sequence number.

3.2 How to Minimize the Information Loss

To minimize the information loss we consider the Distortion metric and it minimize is based on following criteria

- To minimize the Distortion QI have least distortion must be generalize first.
- To minimize the Distortion ,first EQ classes must be merged with in Outer Cluster or 1^{st} level Eq classes, then Eq classes mereged at 1st level and higher based upon having minimum distortion compared to other Eq classes , then inner Clusters that do not satisfy k anonymity merge with the inner Clusters of other Outer Cluster.

3.3 How Records converted to a Equivalence Class and assign to Outer and Inner Cluster

EQ classes both at 0 or 1^{st} level must be generated iterative and integer no is assigned to which starts from 1 in our approach.Eq classes alphabet nos are based number of quasi identifers. One alphabet is assigned to each quasi-identifer.
so size of equivalence class = no of quasi-identifier attributes
Example.
Let a record have values
$\{maritalstatus,\ workclass\} = \{married - civ,\ federalgov\}$
By refering figure 3.1 and 3.2 So its 1^{st} level EQ class generated is 1A 1B
its 0 level EQ class generated is 0A 0C
let other record have values
$\{maritalstatus,\ workclass\} = \{married - af - sp,\ stategov\}$
So its 1^{st} level EQ class generated is 1A 1B

its 0 level EQ class generated is 0B 0E

so both EQ class have different 0 level EQ but same at 1st level EQ

$\{partner - present,\ Gov\}$ In this case both records assign to same outer cluster whose Equivalence class is 1A 1B

But within this outer cluster both records are assign to different inner cluster based upon its 0 level Eq class.

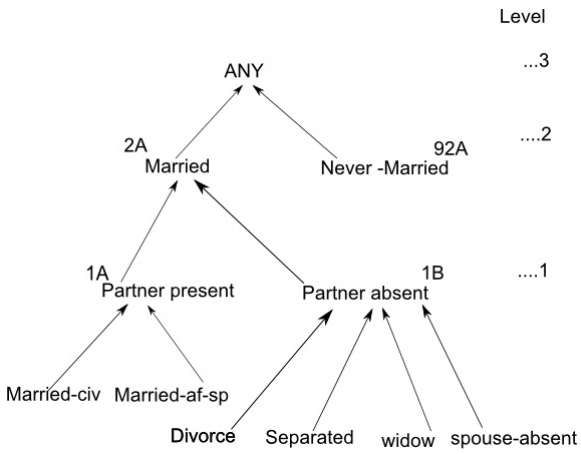

Figure 3.1: Taxonony Tree for quasi-identifer Martial Status

3.4 How to Assign a unique Equivalence Class to Each Cluster at both Level

Equivalence class are generated based upon number of option values at 0 or 1^{st} level, for each quasi identifier. If Equivalence class are generated are generated at 0 level , number of option values are also taken at 0 level, for each quasi identifier. Similarly, if Equivalence class are generated are generated at 1^{st} level, number of option values are also taken at 1st level, for each quasi identifier. Generate String iteratively which is based upon qi value option and assign a integer number to it as its sequence number which is unique for each equivalence class whether 0 or 1^{st} level

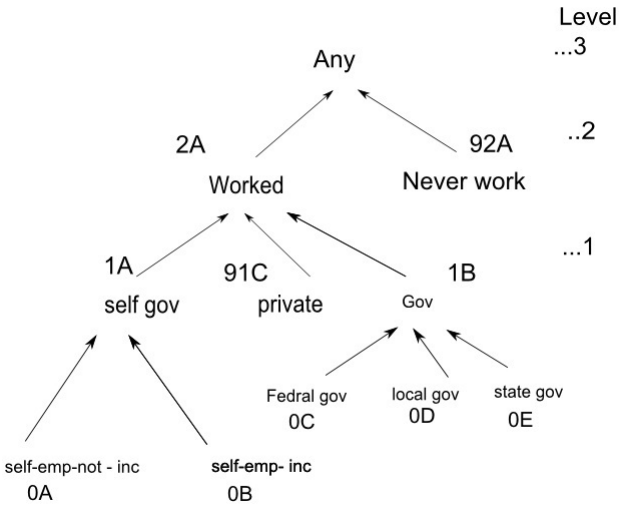

Figure 3.2: Taxonony Tree for quasi-identifer of Workclass

EQ.

3.5 How we can find Equivalence Class Sequence Number in O($|Qi|$)

Example How we can find sequence number of a EQ.

Let for a record equivalence class generated is ACAB

Each alphabet in EQ is corresponding to a QI.

For each Qi , no of different values at 0 or 1 level are the choice (which is to be multiplied) for that QI at 0 or 1 equivalence class .

Step 1

First we have calculate maximum range= ch1 * ch2 *. . .*chn = multiplication of all choices for all Qi

For Quasi identifier no i ,no of choice =ch_i

Step 2

update lower limit=0 and upper limit = maximum range

Here upper range=36,

update divide limit= chi = no of choice for the QIi

For 1^{st} qi alphabet read is A which is the first value for that qi .so it will , choice for qi=$\{A, B\}$

Here, divide limit=2

So upper limit becomes 36/2= 18

here, alphabet read=A and option value= A,

match occurs so it increments the pointer

Step 3

For next Qi again set divide limit=3,choice for qi=$\{A, B, C\}$

Second alphabet of EQ is C

Set range=(upper-lower+1)/divide limit

now, range is =(18-1+1)/3=6

Now set upper=1+6-1=6

We want C but alphabet read is B ,its not matched

So iteration starts

Now,set lower=upper+1;

Lower=6+1=7

Upper=lower+range-1=7+6-1=12

So the second option for QI is B

B $\neq$ C , again mismatch

So iterate again ,

update lower=upper+1=12+1=13

and upper=13+6-1=18

here, alphabet read=C and option value= C,

match occurs so it increments the pointer

Step 4

For 3rd Qi , option values=$\{A, B\}$

update divide limit=2

As it reads first time for this Qi

Set range=(upper-lower+1)/divide limit

range=3

set upper=lower+range-1=13+3-1=15

Next alphabet read is A option value is A

So it increments pointer to next Qi

step 5

For 4^{th} Qi =option values=$\{A, B, C\}$

update divide limit=3

As it reads first time for this Qi

update range=(upper-lower+1)/divide limit

range=(15-13+1)/3=1

upper=lower+range-1=13+1-1=13

Next alphabet read is B option value is A

So mismatch so iteration starts

Update lower= upper+1=13+1=14

Upper=lower+rang-1=14+1-1=14

Now option value is B , next alphabet read is B

So match occurs ,as it is last qi

so It return 14 as equivalence no for ACAB . As it can be observe by refering figure 3.3

3.6 Algorithm to find Cluster Sequence Number

Let QI names: L, M ,N,O

Let us consider EQ at 0 level . Options for each Qi L,M,N,O at 0 level are 2,3,2,3 respectively .So total inner clusters in that outer cluster is their product of option values = 36 .Generate these iteratively based on their option values and assign a integer to it based on its sequence number .

1. AAAA
2. AAAB
3. AAAC
4. AABA
5. AABB
6. AABC
7. ABAA
8. ABAB
9. ABAC
10. ABBA

Algorithm 3 findEqSeqNo ($qidno$,$clusterlevel$,eq[])

1: **for** $i \leftarrow 1, |QI|$ **do**
2: $range \leftarrow range * |QI_i|$ at current clusterlevel
3: **end for**
4: $lower \leftarrow 0$
5: $upper \leftarrow range$
6: **for** $qi \leftarrow 1, |QI|$ **do**
7: $dividelimit \leftarrow optionsof|QI_i|$
8: **for** $pointer \leftarrow 1, |qi_i|$ **do**
9: **if** $pointer$ ==0 **then**
10: $range \leftarrow \frac{upper-lower+1}{dividelimit}$
11: $upper \leftarrow lower + range - 1$
12: **end if**
13: **if** $ValueatQI_i == eq[pointer]$ **then**
14: return upper
15: **else**
16: $lower = upper + 1$
17: $upper = upper + rang$
18: **end if**
19: increment pointer at current qi_i
20: **end for**
21: **end for**

11. ABBB
12. ABBC
13. ACAA
14. ACAB
15. ACAC
16. ACBA
17. ACBB
18. ACBC
19. BAAA
20. BAAB
21. BAAC
22. BABA
23. BABB
24. BABC
25. BBAA
26. BBAB
27. BBAC
28. BBBA
29. BBBB
30. BBBC

31. BCAA
32. BCAB
33. BCAC
34. BCBA
35. BCBB
36. BCBC

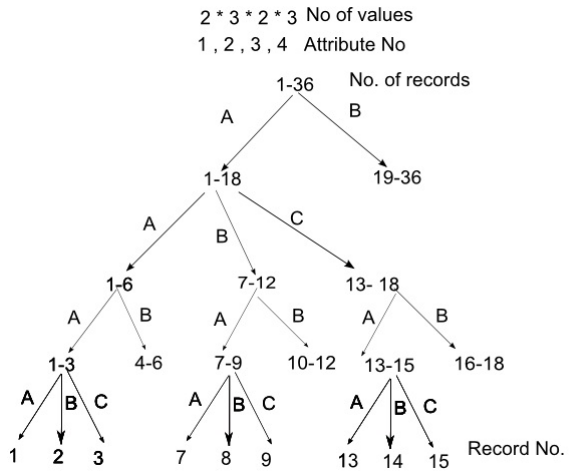

Figure 3.3: Generating Sequence Number for a Equivalence class

3.7 How to find Most Suitable Cluster to Merge

3.7.1 How to find Cluster which are more likely to be merge without linear search

Let 4 qausi-identifier are L M N O taken from dataset . L is left most quasi-identifier while generating equivalence class.Similarly O is right most quasi-identifier.

To generalizing QI: O

(having 3 different values at 0 level cluster) No of values at left side are =0 (NO QI on left of O)

NO of EQ merged :3 =NO of different values for this QI

Simliar EQs are

1,2,3

4,5,6

7, 8,9

10,11,12

13,14,15

...

...

34,35,36

(total different options for Eqs)/ (no of diff. values) = 36/3=12

To generalizing QI: N (having 2 different values at 0 level)

No of values at left side are :3(only 1 QI on left of L)

NO of EQ merged :2 =NO of different values for this QI

Similar EQs are

1,4 (1+3)

2,5

3,6

7,10

8,11

...

....

33,36

(total different options for Eqs)/ (no of diff. values) = 36/2=18

To generalizing QI: M (having 3 different values at 0 level)

No of values at left side are :3*2(QI on left are N ,O)=6

So Skip EQ factor =6

NO of EQ merged :3 =NO of different values for this QI

(total different options for Eqs)/ (no of diff. values) = 36/3=12

For Eq 1,next same merged EQ (1+6)=7,

further next merged EQ = 7+ 6 =13

Similar EQs are

1,7,13

2, 8, 14

3,9,15

. . .

6,12,18

As up to EQ no 18 merged already so increment the pointer to the EQ and start merging in same manner

For Eq 19,next same merged EQ (19+6)=25,

further next merged EQ = 25+ 6 =31

So the next same merged EQ are

19,25,31

20, 26,32

21,27,33

22,28,34

23,29,35

24,30,36

3.7.2 How records can be searched faster than linear search

Explaination

After merging inner clusters with in its the outer cluster in which they all are assigned , if still they are not able to statisfy k anonymity value . It means inner cluster must to be merged with the inner clusters of some other clusters. So we have to search which are the most probable outer cluster whose unmerged inner cluster can be merged with these inner cluster and may satisfy can value. As we have discussed how outer level cluster will be searched it is based upon quasi -identifer which is chosen for generalization. The Quasi-identifier will give minimum distortion at present state of generalized data set must be chose for generalization.
In starting , values must be initialize to
Example :
Upper limit = limits based upon its maximum no of inner cluster present in that cluster
lower limit =0
Starting from the first qusi-identifer ,check whether they match or not
if they match move pointer to next quasi-identifer algo.
if not then use searchlimit Algorithm for this quasi-identifier .search limit use the concept of binary search to reduce time complexity.
For Each attribute based upon its values lower limit increase and upper limit decreases using searchlimit function.
For Next QI these updated limits are lower and upper limits.
Iterate this till the last quasi-identifer .Finally this technique will reduce the searching of similar cluster so it will give less time to search instead of linear search .
Let us take an Example
Let inner Cluster having EQ(0B 0C 0A 1B) of outer cluster whose EQ is (1A 1B 1A 1B) to be merged with inner cluster of Other outer cluster whose EQ is (1A 1B 1B 1B)

So it takes first qi of and see as it is 0B it try to match with EQ no 1 which 0A 0C 0E 1B, mismatch occurs

so it search at half index ie 7.

Set lower limit =7, and again search until it reach to EQ no 12 ie 0B 0C 0F 1B .

so it move pointer to next quasi-identifer to further decrease the search region ,at last it get lower =12 and upper limit is decreased to 14.so its difference is(14-12+1)= 3 cluster to serach linearly instead of searching all inner clusters of outer cluster whose EQ:1A 1B 1B 1B.

Outer Cluster1		Outer Cluster2	
EQ:1A 1B 1A 1B		EQ: 1A 1B 1B 1B	
Inner Cluster	Records	Inner Cluster	Records
0A 0C 0A 1B	5	0A 0C 0E 1B	3
0A 0D 0B 1B	3	0A 0C 0G 1B	4
0A 0D 0C 1B	4	0A 0C 0H 1B	4
0A 0D 0D 1B	2	0A 0D 0E 1B	1
0A 0E 0A 1B	2	0A 0D 0G 1B	2
0A 0F 0C 1B	5	0A 0F 0I 1B	1
0A 0F 0D 1B	4	0B 0C 0F 1B	5
0B 0C 0A 1B	3	0B 0C 0H 1B	2
0B 0D 0A 1B	1	0B 0C 0E 1B	1
0B 0D 0B 1B	4	0B 0D 0E 1B	1
0B 0D 0B 1B	4	0B 0D 0G 1B	1
0B 0D 0C 1B	6	0B 0D 0I 1B	1
0B 0D 0C 1B	6		
0B 0E 0C 1B	1		
0B 0E 0D 1B	8		
0B 0F 0B 1B	3		

3.8 Algorithms used to Anonymize the Original Data

In this section we expalined all the algorithm used , algorithm findEqseqno is used in assignEqclass function to assign the inner and outer cluster number for each record. GetSimilarCluster (qidno,clusterlevel) used in both generlizewithinOuterCluster and generlizeOtherOuterCluster where as searchlimit is used only in generlizeOtherOuterCluster.

3.8.1 Main Algorithm

Algorithm 4 MainAlgorithm (dataset,noOfQIs,k)

1: readfile($origionaldatafile$) and store it in array $data$
2: generateEQclass($noOfQIs, hierarchyfile$)
3: assignEQclasses($data$)
4: **for** $j \leftarrow 1, OuterClusters$ **do**
5: **while** $|innerLevelCluster| < k$ **do**
6: generlizewithinOuterCluster(j)
7: **end while**
8: **end for**
9: **for** $j \leftarrow 1, OuterClusters$ **do**
10: **while** $|innerLevelCluster| < k$ **do**
11: generlizeOtherOuterCluster(j)
12: **end while**
13: **end for**

3.8.2 Alogrithm to generalize with in Outer Level Cluster

Before starting this algorithm, QIs must be sorted based upon their distortion value at current level.QI gives least distortion will be chosen first to generalize.

Algorithm 5 generalizationinOuterCluster(*clusterno*)

```
for i ← 1, noofQi do
    for j ← 1, allInnerCluster do
        if |innerCluster| < k then
            SimilarNumbers = getSimilarCluster(i, 1)
            generalize SimilarNumbers
            if SimilarNumbers[j]size ≥ k then
                store into final Eqclass
            end if
        end if
    end for
end for
```

3.8.3 Algorithm used to decrease the searchlimits

Inner cluster when merge with inner cluster of differnet outer cluster instead of using linear search searchlimit function is used to decrease number of cluster which is to be the searched.

Algorithm 6 searchlimits (*qidno*,*qidnovalue*,*limits* [ ])

1: $lowerlimit \leftarrow limits[0]$
2: $upperlimit \leftarrow limits[1]$
3: **while** $qidva > EQrowno[lowerlimit][qidno]$ **do**
4: $\quad lowerlimit \leftarrow \frac{(lowerlimit+upperlimit)}{2}$
5: **end while**
6: **while** $qidvalue > EQrowno[lowerlimit][qidno] \& lower > 1$ **do**
7: $\quad lowerlimit - -$
8: **end while**
9: **while** $qidvalue < EQrowno[lowerlimit][qidno]$ **do**
10: $\quad upperlimit \leftarrow \frac{upperlimit}{2}$
11: **end while**
12: **while** $qidvalue < EQrowno[lowerlimit][qidno] \& lower > 1$ **do**
13: $\quad upperlimit - -$
14: **end while**
15: $limit[0] \leftarrow lowerlimit$
16: $limit[1] \leftarrow lowerlimit$

3.8.4 Algorithm to find the suitable Cluster to mmerge

It is used to find the most suitable cluster to merge with a cluster without using linear search .It is used in both outer and inner level cluster .

Algorithm 7 GetSimilarCluster (*qidno*,*clusterlevel*)

1: **for** $i \leftarrow 1, QI$ **do**
2: $totalcluster \leftarrow totalcluster * |QI_i|$ at current clusterlevel
3: **end for**
4: $mergingLimit \leftarrow |QI_{qidno}|$at current clusterlevel
5: $reducedSize \leftarrow \frac{totalcluster}{mergingLimit}$
6: int SimilarNumbers[reducedSize][mergingLimit]
7: $leftSideOptions \leftarrow findLeftSideOption(qidno)$
8: **for** $i \leftarrow 1, reducedSize$ **do**
9: **for** $j \leftarrow 1, merginglimit$ **do**
10: $EQNo \leftarrow EQNo + LeftSideOptions$
11: $SimilarNumbers[i][j] \leftarrow EQNo$
12: **end for**
13: **if** *qidno* $\neq$Last Quasi Identifier **then**
14: $EQNo \leftarrow EQNo + 1$
15: **else**
16: $EQNo \leftarrow EQNo + merginglimits$
17: **end if**
18: **end for**
19: return SimilarNumbers

Chapter 4

Experiment Results

4.1 Implementation Environment and Data Set

Implementation is done on System having configuration Dual core 2.0GHz , 2.5GB RAM. Our Implementation is done on Java Platform.Complete Adult Data Set which contains 32,561 records is taken for analysis results.The attributes for quasi identifier are Age which is numeric, Work class which is categorical, Education which is categorical, Marital status is categorical, race which is categorical, gender is categorical, Occupation and salary are sensitive attributes.

4.2 Evaluation Metrics

We have taken Distortion, Discernibility Metricand Exceution Time as parameters to evaluate and analyse the result for k values taken as 2, 5, 10 .

4.2.1 Distortion Metric

To measure the information loss of anonymized Data , we calculated Distortion Metrics at K = 2, 5, 10. By refering figures 4.2 ,4.5 , 4.8 we can conclude that when k is not so large,k= 2 ,5 our Approach give lesser distortion than KACA and Top-

S.No	Attributes	Generalizations	Distinct Value	Height
1	Work Class	Taxonomy Tree	7	3
2	Education	Taxonomy Tree	16	4
3	Marital Status	Taxonomy Tree	7	3
4	Race	Taxonomy Tree	5	2
5	Sex	Suppression	2	1
6	Occupation	Taxonomy Tree	14	2
7	Salary	Suppression	2	1

Table 4.1: Description of Adult Dataset

Down Algorithm but when k is large, k=10 our approach give little more distortio or information loss than other algorithms.

4.2.2 Execution Time

We considered Execution time also to evaluate and compare our approach with KACA and TopDown-KACA.By refering figures 4.1 , 4.4 , 4.7 we can conclude that for all k values 2, 5, 10 and our approach take lesser execution time than TopDown-KACA and KACA algorithm. For all k values taken and for all number of quasi identifier taken so we can conclude our approach is faster compared to others.

4.2.3 Discernibility Metric

We used Discernibility Metric to measure the quality of anonymized data , the lesser is discerniblity cost ,better is the quality is anonymized Data . By refering figures 4.3 ,4.6 , 4.9 we can conclude that For smaller K value k=2,5 and , for all number quasi identifers taken our approach give better anonymized data than KACA and TopDown -KACA algorithm and if K is large, K= 10 and number of quasi identifier taken not large our approach gives lesser discernibility otherwise gives similar result.

4.2.4 Plotted Results

For K value = 2 , calculated metrics Execution time vs QI ,Distortion vs Quasi-identifier , Discernibility vs Quasi-Identifier are plotted in figures 4.1 ,4.3 , 4.2 respectively.

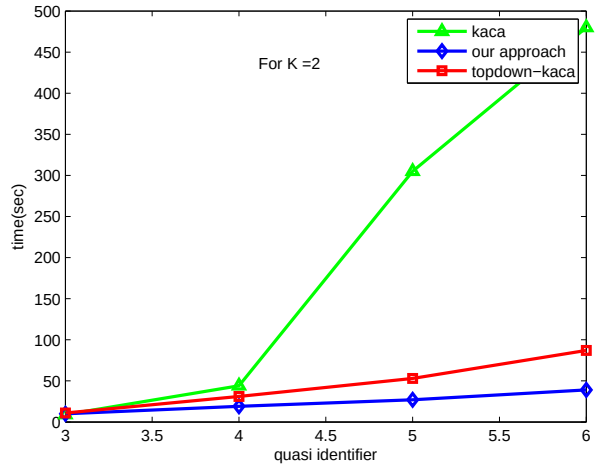

Figure 4.1: Execution Time(sec) vs Quasi-Identifier

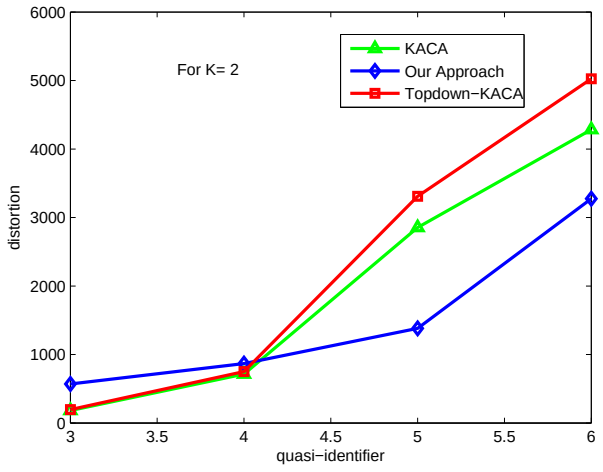

Figure 4.2: Distortion vs Quasi-Identifier

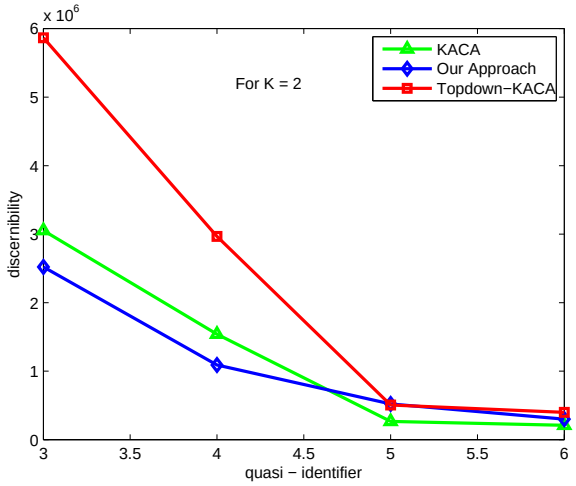

Figure 4.3: Discernibility vs Quasi-Identifier

For K value = 5 , calculated metrics Execution Time vs QI ,Distortion vs Quasi-identifier , Discernibility vs Quasi-identifier are plotted in figures 4.4 ,4.6 , 4.5 respectively.

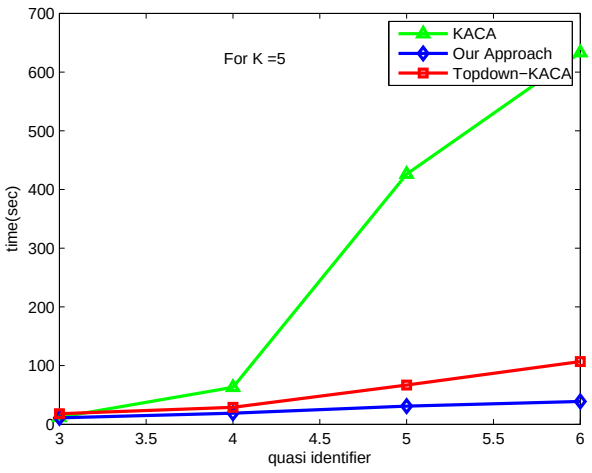

Figure 4.4: Execution Time(sec) vs Quasi-Identifier

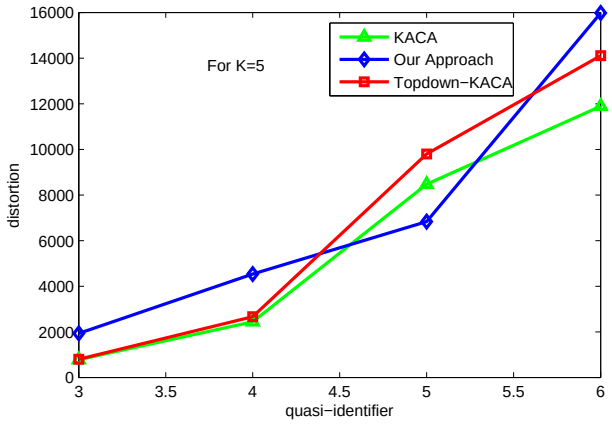

Figure 4.5: Distortion vs Quasi-Identifier

For K value = 10 , calculated metrics Execution Time vs QI ,Distortion vs Quasi-identifier , Discernibility vs Quasi-identifier are plotted in figures 4.4 ,4.6 , 4.5 respectively.

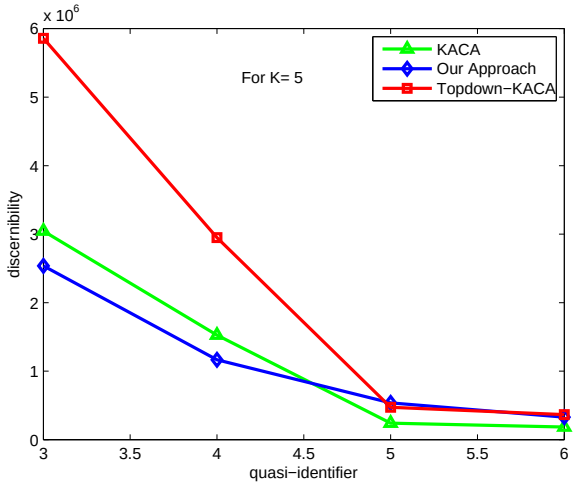

Figure 4.6: Discernibility vs Quasi-Identifier

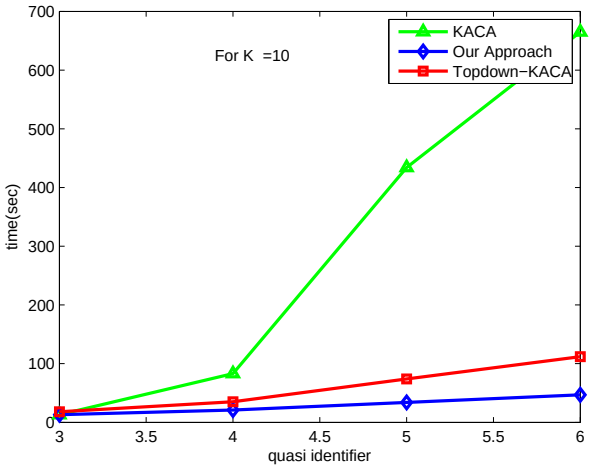

Figure 4.7: Execution Time(sec) vs Quasi-Identifier

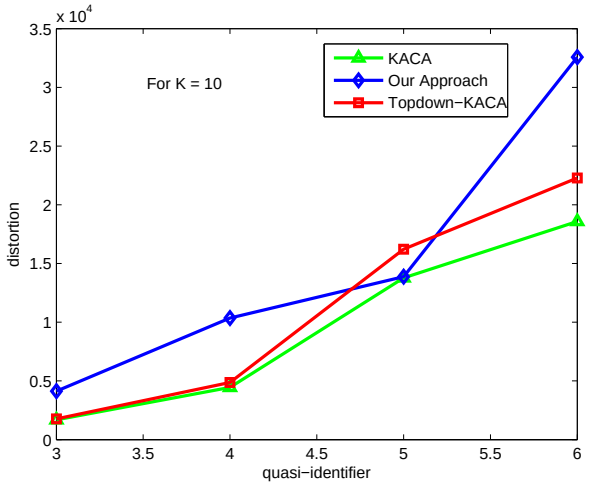

Figure 4.8: Distortion vs Quasi-Identifier

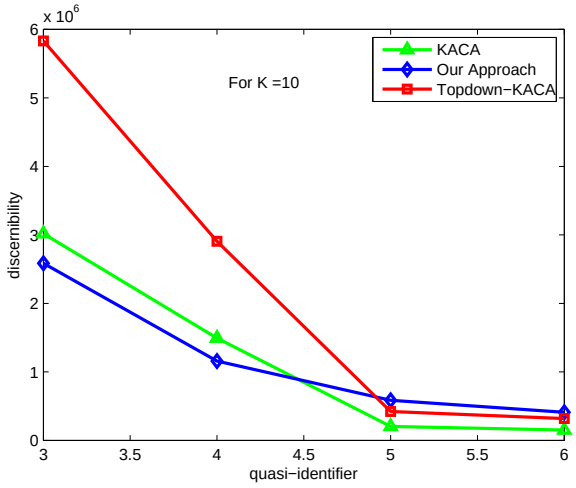

Figure 4.9: discernibility vs quasi-identifier

Chapter 5

Conclusion

Local Recoding Algorithm gives lesser information loss compared to Global Recoding but they takes much more time to execute compared to global recoding and complicated to implement compared to Global Recoding Algorithms. Local Recoding Algorithm Execution time mostly depends on how most suitable clusters can be searched to merge them for satisfying k value, linear search takes much time. In this searching it also have to search some clusters which are completely different and not suitable to merge which can be skipped to search by partition the database into some bigger clusters as this technique is implemented in Topdown-KACA, instead of linear search. As our approach find the most suitable clusters to merge without using linear search based on the mathematical relation between their Equivalence class which is uniquely assigned to them while considering distortion metric also to minimum the information loss. Our Purposed Algorithm takes lesser time to execute compared to KACA and TopDown-KACA almost half and while other metric such discernibility and distortion also give better results for most of the number of quasi-identifiers.TopDown-KACA takes less execution time compared to KACA but its information loss and discernibility metrics give lower quality result compared than KACA. It can be used to implement l-diversity also, which is the next level of privacy .

Bibliography

[1] A. Machanavajjhala, D. Kifer, J. Abowd, J. Gehrke, and Vilhuber. Privacy: Theory meets practice on the map. In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*, pages 277–286. IEEE, 2008.

[2] Z. Yang, S. Zhong, and R. N. Wright. Anonymity-preserving data collection. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 334–343, 2005.

[3] J. Yu, J. Han, J. Chen, and Z. Xia. Topdown-kaca: An efficient local-recoding algorithm for k-anonymity. In *GrC*, pages 727–732. IEEE, 2009.

[4] B. C. M. Fung, K. Wang, R. Chen, and P. S. Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys*, 42(4), 2010.

[5] Q. Tang, Y. Wu, S. Liao, and X. Wang. Utility-based k-anonymization. In *Proceeding - 6th International Conference on Networked Computing and Advanced Information Management, NCM 2010*, pages 318–323, 2010.

[6] R. C. Wong, J. Li, A. W. Fu, and K. Wang. (α, k)-anonymity: An enhanced k-anonymity model for privacy-preserving data publishing. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, volume 2006, pages 754–759, 2006.

[7] P. Samarati. Protecting respondents' identities in microdata release. *IEEE Transactions on Knowledge and Data Engineering*, 13(6):1010–1027, 2001.

[8] L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowlege-Based Systems*, 10(5):557–570, 2002.

[9] R. J. Bayardo and R. Agrawal. Data privacy through optimal k-anonymization. In *Proceedings - International Conference on Data Engineering*, pages 217–228, 2005. Cited By (since 1996):247.

[10] Jiuyong Li, R.C.W Wong, Ada Wai-Chee Fu, and Jian Pei. Anonymization by local recoding in data with attribute hierarchical taxonomies. *Knowledge and Data Engineering, IEEE Transactions on*, 20(9):1181–1194, 2008.

[11] P. C. Chu. Cell suppression methodology: The importance of suppressing marginal totals. *IEEE Transactions on Knowledge and Data Engineering*, 9(4):513–523, 1997. Cited By (since 1996):4.

[12] C. Dwork and A. Smith. Differential privacy for statistics: What we know and what we want to learn. *Journal of Privacy and Confidentiality*, 1(2):2, 2010.

[13] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3, 2007.

[14] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical report, Technical report, SRI International, 1998.

[15] K. Nissim. Private data analysis via output perturbation. In *Privacy-Preserving Data Mining*, pages 383–414. Springer, 2008.

[16] F. McSherry and I. Mironov. Differentially private recommender systems: building privacy into the net. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 627–636. ACM, 2009.

[17] T. Dalenius. Finding a needle in a haystack-or identifying anonymous census record. *Journal of official statistics*, 2(3):329–336, 1986.

[18] D. Kifer and J. Gehrke. Injecting utility into anonymized datasets. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 217–228. ACM, 2006.

[19] L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[20] B. C. M. Fung, K. Wang, and P. S. Yu. Top-down specialization for information and privacy preservation. In *Data Engineering, 2005. ICDE 2005. Proceedings. 21st International Conference on*, pages 205–216. IEEE, 2005.

[21] B. C .M. Fung, K. Wang, and P. S. Yu. Anonymizing classification data for privacy preservation. *Knowledge and Data Engineering, IEEE Transactions on*, 19(5):711–725, 2007.

[22] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Incognito: Efficient full-domain k-anonymity. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 49–60. ACM, 2005.

[23] R. C. W. Wong, J. Li, A. W. C. Fu, and K. Wang. (α, k)-anonymity: an enhanced k-anonymity model for privacy preserving data publishing. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 754–759. ACM, 2006.

[24] V. S. Iyengar. Transforming data to satisfy privacy constraints. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 279–288. ACM, 2002.

[25] J. Li, R.C.W. Wong, A. W. C. Fu, and J. Pei. Achieving k-anonymity by clustering in attribute hierarchical structures. In *Data Warehousing and Knowledge Discovery*, pages 405–416. Springer, 2006.

[26] X. Xiao and Y. Tao. Anatomy: Simple and effective privacy preservation. In *Proceedings of the 32nd international conference on Very large data bases*, pages 139–150. VLDB Endowment, 2006.

[27] X. Xiao and Y. Tao. Personalized privacy preservation. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 229–240. ACM, 2006.

[28] K. Wang, B.C.M. Fung, and S. Y. Philip. Handicapping attacker's confidence: an alternative to k-anonymization. *Knowledge and Information Systems*, 11(3):345–368, 2007.